

SYSTEM AND METHOD FOR NAME RESOLUTION

Field of the Invention

5 The present invention relates to publishing documents and information stored on individual machines, or nodes, in a distributed network and sharing these documents with other nodes in the network. More particularly, the present invention relates to systems and methods for providing another user access to such published documents.

Background of the Invention

10

In distributed networks or systems having many connected nodes or processes, the ability to share files, documents, and information between nodes of the network is important.

There are several conventional ways that users may share files and documents, including posting the documents on web servers, where access may be open to anyone or restricted to pre-approved users by a password protection system. The problem with this system is two-fold. First, it typically requires a third party, such as a web-hosting service, to maintain the documents on a server computer, which generally requires an initial set up fee and recurring maintenance fees. Second, the involvement of a third party inherently reduces the user's control over the files or documents stored on the web server.

15
20

Another way to share files among users of a distributed network is for a user who wishes to share files ("the publisher") with another user ("the accesser") to provide the accesser with his or her machine location, and a path name on the machine where the documents are stored. The accesser accesses the publisher's documents by entering the machine location and path name into a web browser or other access program to reach the publisher's machine. In this situation, the publisher's machine typically provides a security system that permits access only to approved accessing users.

25

In order to enter the machine location, the user must know the same. One method of entering the machine location involves entering an Internet Protocol ("IP")

30

address. An "IP Address" is a unique string of numbers that identifies a computer on the Internet. IP addresses contain 32-bits, organized into four sets of three-digit numbers between 0 and 255, which are separated by periods, like this: 123.123.023.002. All machines on the Internet must have an IP address and no two computer systems
5 may have the same IP address at the same time. IP addresses may be dynamic or static. A static IP address is one that is permanently assigned to a computer system – it is the only IP address used by that system. A dynamic IP address is one that is assigned on the fly from a group of IP addresses assigned to an organization, for example. Although
10 no IP address can be used for two computer systems at the same time, each computer system may use a multitude of different IP addresses. IP addresses are not user-friendly because they contain only numbers with no readily understandable meaning. For this reason, it is nearly impossible for people to remember their own IP addresses, let alone the IP addresses of machines belonging to other people.

The Domain Name System (DNS) makes it easier to find machine locations by
15 allowing a familiar string of letters (the "domain name") to be used instead of the arcane IP address. So instead of remembering and typing 66.201.69.207, users can type www.microsoft.com. Domain names are also used for reaching e-mail addresses and for other Internet applications. The domains names are resolved, i.e., converted from the domain name to the IP address, through a service hosted on a number of servers
20 located throughout the Internet. However, DNS has at least two limitations. First, it requires the use of third party equipment, the DNS servers, to handle the name resolution. Second, it is not very secure because almost anyone can determine a specific machine name and location for any computer system. Therefore, the conventional DNS is not a very effective method for publishing resources on a
25 computer system.

Another way to get a machine location is to use a public key. Public keys are associated with a particular person and consist of a long string of bytes, for example 32 numbers and letters, such as KP12JSP2345L1298FE23K1KSERQOC38S. Public keys are typically used to enable users of an unsecure public network, such as the Internet, to
30 securely and privately exchange data through the use of a public and a private

cryptographic key pair that is obtained and shared through a trusted authority. Public key cryptography is the most common method on the Internet for authenticating a message sender or encrypting a message. Public keys may be obtained in a number of ways. For example, a public key may be created and assigned by an operating system
5 when a person creates a user account on a computer system. Alternatively, in public key cryptography, a public key and private key are created simultaneously using the same algorithm by a certificate authority.

Public keys are connected to machine locations through use of the peer name resolution protocol ("PNRP"), which stores each person's public key with their current
10 location in a publicly available table or some other searchable data structure. Thus, if you know a person's public key, one may use PNRP to determine its associated current machine location, which is usually in the form of an IP address. However, like an IP address, a public key is a string of many bits that have no meaning and even harder to memorize than the IP address itself.

15 Yet another way to determine a machine location involves receiving (such as through email) a link that incorporates or contains the machine location. When the link is selected, the browser automatically enters the IP address or public key associated with the link into the browser. Unfortunately however, an accessing principal would need to save the email and find and open it every time the accessing node wanted to
20 access the publishing node's resources. This is time consuming, cumbersome, and wastes storage capacity on the accessing machine.

The invention may be implemented as a computer process, a computing system or as an article of manufacture such as a computer program product or computer readable media. The computer program product may be a computer storage media
25 readable by a computer system and encoding a computer program of instructions for executing a computer process. The computer program product may also be a propagated signal on a carrier readable by a computing system and encoding a computer program of instructions for executing a computer process.

These and various other features as well as advantages, which characterize the present invention, will be apparent from a reading of the following detailed description and a review of the associated drawings.

5

Summary of the Invention

The present invention relates to accessing and publishing documents between two computer systems or nodes that are connected together in a network environment and more particularly relates to a system and method for name resolution between a user-friendly handle signifying identity with a machine location where the documents
10 are stored, so that network users may easily access these documents through knowledge of the user-friendly handle. In other words, a user's documents may be accessed through use of a user-friendly handle identifying the user and a path name that describes the documents of interest, which are resolved to a machine location where the documents are stored. An email address is an exemplary example of a user-friendly
15 handle and an IP address or public key are exemplary examples of machine locations. The computer system or node where the documents are stored is referred to as the publishing computer system or node and the computer system that seeks to access documents stored on the publishing node is referred to as the accessing computer system or node.

20 In accordance with particular aspects, a method of accessing documents stored on a first publishing computer system through a second accessing computer system begins when an identity information document from the first computer system is stored on the second computer system. The identity information document has, at a minimum, a user-friendly handle identifying a principal and a machine location for the first
25 computer system. When an initial request for access to documents containing the user-friendly handle is made, it is intercepted by the second computer system. The second computer system replaces the user-friendly handle with the machine location and sends the amended request to the machine location of the first computer system. This method allows the second computer system to access documents on the first computer system

with knowledge only on the user-friendly handle. Knowledge of the machine location, IP address, or public key for the first computer system is not necessary.

In accordance with other aspects, the present invention relates to a method of publishing documents between a plurality of nodes connected in a network
5 environment. The method begins by sending an identity information document from a publishing node to an accessing node. The identity information document includes at least a user-friendly handle identifying a principal and a machine location for the publishing node. The identity information document is thereafter stored on the
10 accessing node. When an initial request for access to documents is made from the accessing node to the publishing node, the user-friendly handle is resolved with the machine location. Next, the amended request for access to documents is sent from the accessing node to the machine location of the publishing node.

In accordance with still other aspects, the present invention relates to a computer system comprising a storage module for storing an identity information document
15 received from a second computer system and a name resolution module connected to the storage module. The name resolution module intercepts requests for access to documents stored at the user-friendly handle of the identity information document and amends the request to replace the user-friendly handle with the machine location from the identity information document.

20 In accordance with still other aspects, the present invention relates a computer readable medium encoding a computer program of instructions for executing a computer process for name resolution. The process starts with a store operation storing an identity information document from a publishing computer system. Next an
intercept operation intercepts an initial request for access to documents stored on the
25 publishing computer system when the initial request contains the user-friendly handle. Last, an amend operation amends the request to replace the user-friendly handle with the machine location.

Brief Description of the Drawings

Fig. 1 illustrates a communication or distributed network of nodes that incorporates aspects of the present invention.

Fig. 2 illustrates a computer system that may be used according to particular
5 aspects of the present invention.

Fig. 3 illustrates the structure of an identity information documents according to particular aspects of the present invention.

Fig. 4 illustrates a representation of the software environment according to aspects of the present invention.

10 Fig. 5 illustrates a flow chart of operational characteristics of the present invention with respect to an accessing node.

Fig. 6 illustrates a flow chart of operational characteristics of an alternative embodiment of the present invention with respect an accessing node and a publishing node.

15 Fig. 7 illustrates a flow chart of operational characteristics of yet alternative embodiment of the present invention with respect an accessing node and a publishing node.

Detailed Description of the Preferred Embodiment

20 A distributed environment 100 incorporating aspects of the present invention is shown in Fig. 1. The environment 100 has at least one computer system 102 and potentially other computer systems such as 108, 110, and 152, wherein the various computer systems are referred to as “nodes” or “machines.” As used herein, a “computer system” shall be construed broadly and is defined as “one or more devices or
25 machines that execute programs for displaying and manipulating text, graphics, symbols, audio, video, and/or numbers.” Nodes in the network may be any type of computer system, including, without limitation, a telephone such as node 108, a PDA such as node 110, a desktop computer such as nodes 102 and 152, a lap top computer (not shown), and many others. Further, although shown as computer systems, nodes
30 102, 108, 110 and 152 may alternatively be computer processes within a computer

system. Alternatively, the nodes 102, 108, 110 and 152 may combine a combination of separate computer systems distributed across a local area network, a wide area network, or a combination of separate network communications.

As stated, each of the computer systems 102, 108, 110 and 152 are considered
5 nodes within the environment 100 capable of communicating with other nodes within the environment 100 and sharing documents, information, and resources with other network nodes. Furthermore, the nodes may communicate via separate protocols such as TCP/IP or other network and/or communication protocols, implemented over networks such as the Internet 106. That is, although shown as connected by seemingly
10 direct arrows, the separate nodes 102, 108, 110 and 152 may in fact be in communication with other nodes via other indirect ways. Indeed, the connections shown in 100 merely indicate that a node may communicate with another node.

Communication between machines 102, 108, 110 and 152 may be achieved, as stated, by many communication protocols. The definition of a communication used
15 herein relates to the transfer of a message, an event, or any other information from one node to another. In an embodiment, the nodes of environment 100 may be able to communicate with all other nodes in the network 100, but such a requirement is not necessary. In order to communicate information from a first node to a second node, the first node needs the machine location or some other identifying information for the
20 accessing node. Using the machine location, the sending node may send the information using any transfer protocol.

Although only four nodes 102, 108, 110 and 152 are shown in Fig. 1, the network environment may include other nodes. Indeed, the number of nodes for environment 100 may be quite extensive incorporating thousands, to tens of thousands
25 of nodes or more. Hence, the present invention is beneficial in scaling the environment 100 as needed so that practically any number of nodes may communicate information according to the present invention.

The present invention relates to a user-friendly system and method for publishing or sharing resources stored on one network node, the "publishing node,"
30 with another network node, the "accessing node." Computer system 102 is an example

of a publishing node and includes a database 104 having data organized into one or more directories, or folders, such as folders 118, 120, 122, and 124. The database 104 relates to a generic file system or other organized data system for storing and retrieving electronic documents. As such, the database 104 may include any type of data or file, which will be referred to herein as “documents.” The term “documents” should be construed broadly and may include, but is not limited to, photographs, video clips, audio clips, text files, presentations, software code, or any other personal resource stored on a computer system. The documents may be organized within the database in any way, including without limitation, in folders and subfolders with descriptive names. For example, folder 118 may contain “jpeg” files and be entitled “photos,” folder 120 may contain audio clips and be labeled music, folder 122 may contain “mpeg” files and be labeled video, and folder 124 may contain executable code and be labeled “software.” The ellipse 126 indicates that there may be generally any number of folders in database 104 containing any types of documents.

Although only one database 104 is shown, ellipse 128 indicates that the machine 102 may have more than one database that is organized in the same way or a different way than database 104. For example, the database 104 may be specific to a principal user of the machine and stored in that principal’s profile located in the machine 102. As used herein, a “principal” should be construed broadly and defined as any entity capable of acting digitally. Principals include without limitation individual people, groups or sets of people meaning individuals, households, organizations, explicit groups, and people in common roles or people who share attributes of some kind as well as the various electronic devices through which these individuals act. Another principal may have a different database stored in machine 102 that is only accessible by that principal.

Computer system 102 additionally maintains a set of self-identity information 130 that comprises a variety of information about the principal represented by or using the computer system 102. This information, for example, may include a name, email address, website URL, physical mailing address, machine location for the principal’s computer system, and other personal information as well as a usage policy describing how this information may be used. Each of these different, identifying elements will be

referred to hereafter as identity claims. Importantly, the set of identity claims include at least a user-friendly handle identifying the computer system 102 and a machine location.

Computer system 102 is capable of creating an identity information document 116 containing some or all of the self-identity information 130 and sending the identity information document 116 to any other node in the environment 100 as shown by the dashed arrows 132 in Fig. 1. As used herein, "identity information document" shall mean a subset of identity information for a principal transmitted from one machine to another so as to allow the device that receives the identity information document to recognize the principal and the principal's associated digital events. Details of one possible format for the identity information document 116 will be discussed below with reference to FIG. 3. However, generally speaking, the identity information document 116 may be in a format suitable for transferring information between disparate systems across various types of channels. The channel used to transfer the identity information document 116 from the computer system 102 to a receiving system, such as computer system 152, can be any of a variety of possible media. For example, email, instant messaging, beaming, and many other mechanisms may be used as channels. Further, the channel may or may not be secure.

The computer system 152 is an example of a accessing node and contains a control module 154 that reads the incoming identity information document 116 and accepts it or rejects it depending upon different variables. For example, if the identity information document 116 originates from a known principal, the computer system 152 will accept and store the identity information document 116. However, if an identity information document 116 arrives from an unknown principal, or if there is a fear that impostors have sufficient motivation to open and modify or forge the identity information document 152, the computer system 152 may reject the identity information document 116 or seek further verification of its authenticity.

In an embodiment, once the identity information document 116 is accepted, the identity claims it contains are added to a recognized identity information database 156 of the computer system 152, which may use this information to verify and authenticate

the computer system 102 in the future and employ channels of interacting with that principal that may not otherwise be trusted. The principal represented by the identity information document 116 may then, for example, be verified and given access to resources on the computer system 152 such as documents stored in a database like the database 104 of the computer system 102.

Additionally, after the computer system 152 accepts and stores the identity information document 116 in its database 156, it will be able to use the identity claims for the principal of the computer system 102 to more easily and quickly access the documents contained in the database 104 of the computer system 102 as will be explained in detail with reference to FIGS. 4-6.

In general however, the computer system 152 has a resolution module 160 that intercepts requests to access documents on computer system 102, which requests may come from a principal using the computer system 152 or may happen automatically. The initial request from the principal includes a user-friendly handle, e.g., the user friendly handle identity claim from the computer system 102. The resolution module 160 translates the user-friendly handle into a machine location, also received from the computer system 102, and enters the appropriate information into the web browser of the computer system 152, which, in turn, accesses the requested data on the computer system 102. Because of the resolution operation of the resolution module 160, the principal of the computer system 152 does not have to remember the machine location, IP address or public key for the computer system 102, but instead needs only to remember the user-friendly handle (and an associated path).

In one embodiment of the present invention, computer system 152 also contains self-identity information (not shown) and the computer system 102 also contains a control module (not shown) and a recognized identity database (not shown). In order for the principal of computer system 152 to access the documents in the database 104 of computer system 102, the principal of computer system 152 must send its own identity information document 158 to computer system 102 before he or she will be given access to documents in database 104. In other words, there must be a mutual exchange of identity information documents 116 and 158 for a principal of computer system 152

to easily request and gain access to the documents in the database 104. Alternatively, the computer system 102 may not require any verification or authentication process and may allow any accessing principal to access the documents contained in the database 104. In this case, only computer system 102 would need to send the identity

5 information document 116 to computer system 152 in order for system 152 to access the database 104. That is, only a one-way exchange of identity information documents, from the publishing node to the accessing node, is needed.

Fig. 2 shows a computer system 200 that may represent one of the nodes, such as 102 or 152 shown in Fig. 1, which receives and disseminates information and
10 publishes and shares documents in accordance with the present invention. The system 200 has at least one processor 202 and a memory 204. The processor 202 uses memory 204 to store documents in databases, such as database 104, self-identity information 130, and recognized identity database 156.

In its most basic configuration, computing system 200 is illustrated in Fig. 2 by
15 dashed line 206. Additionally, system 200 may also include additional storage (removable and/or non-removable) including, but not limited to, magnetic or optical disks or tape. Such additional storage is illustrated in Fig. 2 by removable storage 208 and non-removable storage 210. Computer storage media includes volatile and nonvolatile, removable and non-removable media implemented in any method or
20 technology for storage of information such as computer readable instructions, data structures, program modules or other data. Memory 204, removable storage 208 and non-removable storage 210 are all examples of computer storage media. Computer storage media includes, but is not limited to, RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM, digital versatile disks (DVD) or other optical
25 storage, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to store the desired information and which can be accessed by system 200. Any such computer storage media may be part of system 200. Depending on the configuration and type of computing device, memory 204 may be volatile, non-volatile or some combination of
30 the two.

System 200 may also contain communications connection(s) 212 that allow the device to communicate with other devices, such as other nodes 108, 110 or 152 shown in Fig. 1. Additionally, system 200 may have input device(s) 214 such as keyboard, mouse, pen, voice input device, touch input device, etc. Output device(s) 216 such as a display, speakers, printer, etc. may also be included. All these devices are well known in the art and need not be discussed at length here.

Computer system 200 typically includes at least some form of computer readable media. Computer readable media can be any available media that can be accessed by system 200. By way of example, and not limitation, computer readable media may comprise computer storage media and communication media. Computer storage media has been described above. Communication media typically embodies computer readable instructions, data structures, program modules or other data in a modulated data signal such as a carrier wave or other transport mechanism and includes any information delivery media. The term "modulated data signal" means a signal that has one or more of its characteristics set or changed in such a manner as to encode information in the signal. By way of example, and not limitation, communication media includes wired media such as a wired network or direct-wired connection, and wireless media such as acoustic, RF, infrared and other wireless media. Combinations of any of the above should also be included within the scope of computer readable media.

Fig. 3 illustrates an exemplary format for an identity information document 300 that may represent one or both of the identity information documents 116 and 158 shown in Fig. 1. As a data structure, the identity information document 300 is a collection of identity claims and other attribute/property claims that may be bound to a public key and governed by an embedded use policy. XML may be used as the encoding language for the identity information document. However, other formats are considered equally suitable. The elements of the identity information document 300 may also be optionally encrypted if they contain information whose confidentiality must be maintained.

The data within the identity information document 300 can be divided into two categories, including a set of logical components 302 and a set of attribute tags 316.

The identity information document 300 has six principal logical components: 1) a principal identifier 304; 2) one or more identity claims of the principal 306; 3) a display
5 name and zero or more selectively disclosed attributes of the principal 308; 4) one or more keys 310 for the principal enveloped in any acceptable formats (for example public keys in X509v3 certificates); 5) a use policy that expresses the principal's privacy requirements 312; and 6) a digital signature over the entire content of the identity information that protects the integrity of the data and authenticates the sender in
10 the case of identity information updates 314. Each of these six logical components 302 are discussed in turn below.

The principal identifier 304 is a user-friendly handle that identifies the principal that is the principal of the identity claims contained in the identity information document 300. The preferred principal identifier 304 is the principal's email address if
15 the principal is a person. However, the principal identifier should be construed broadly as any type of user-friendly handle that uniquely identifies a principal and may include, but is not limited to, email addresses, telephone numbers, mobile phone numbers, etc.

Identity claims 306 include additional structured information relating to the principal that is the subject of the identity information document. Identity claims
20 should be construed broadly as descriptive information about a principal that may include, but are not limited to, physical mailing addresses, telephone and facsimile numbers, employer information, date of birth, etc. Even more particularly, "identity claims" are uniquely true about one entity (a person, an organization, etc). Thus, in some instances, a telephone number may be a valid identity claim for one person. For
25 example, a cell phone number, a direct dial work number or a home number may be a valid identity claim for someone who does not share a cell phone, a direct dial work number or who lives alone. In other cases, a telephone may not be a valid identity claim for a single person, such as a home phone number shared by a family of people. In such a case the home telephone number could be an identity claim to represent the
30 household, but not one individual.

The machine location 308 provides a unique address for the principal's computer system, and may include without limitation, an IP address or a public key. Like a physical street address, the machine location is necessary to locate, connect to, and or communicate with a computer system within a distributed network. The

5 machine location 308 may actually include a list of machine locations if a principal has more than one computer system or if the computer system is mobile, such as a laptop computer system or a PDA. In one embodiment, each machine location may contain a list of documents stored at that machine location. For example, if a principal has a first

10 computer system storing jpeg files at a "photos" path name and a second computer system storing sound recordings at a "music" path name, the machine location 308 may include an IP address for each of the first and second computer systems and an indication that "photos" are stored at the machine location for the first computer system and that "music" is stored at the machine location for the second computer system.

The keys portion 310 contains one or more keys, such as public keys

15 encapsulated within a certificate format (for example, X509v3 certificates). The keys 310 can be public keys and can be included in the identity information as recognition information or the subject of the identity information. If a certificate is used, it may be self-signed or issued by a certificate authority.

The use policy 312 conveys the principal's instructions about permissible uses

20 for the contents of the identity information document 300. For example, the use policy 312 may indicate that the contents of the identity information should not be divulged to others. A recipient's recognized identity information database, such as database 156 in Fig. 1, will store the use policy along with the rest of the information defining the principal.

25 The digital signature 314 provides a principal with the ability to sign data within the identity information document. XML signatures have three ways of relating a signature to a document: enveloping, enveloped, and detached. According to one embodiment of the present invention, the identity information document uses XML enveloped signatures when signing the identity information content.

The identity information document 300 can carry six or more attributes tags 316 relating to the identity information document 316 itself. Although not shown, the attribute tags may include an ID value for the identity information document 300, version information for the document 300 and/or the principal type that the document
5 300 represents, e.g., a person, computer or organization. Other attribute tags may also be employed.

In an embodiment, the identity information document is stored in a generalized manner on the primary computer system, such as systems 102 and 402 described above in conjunction with Figs. 1 and 4, respectively.

10 Fig. 4 illustrates the functional components related to accessing and sharing or publishing documents between two or more computer systems 402 and 450. That is, Fig. 4 represents software components or modules according to aspects of the present invention. In particular, Fig. 4 illustrates a publishing node or computer system 402 and an accessing node or computer system 452 and their associated modules used to publish
15 and access documents according to the present invention. Computer system 402 has among other modules that are not shown, the following components: 1) a database 404; 2) an identity information document module 414; 3) a user-interface module 416; 4) a verification and publish module 418; 5) a memory access module 420; 6) a communication module 422; and (vii) a self-identity information database 424.

20 Like the database 104 shown in Fig. 1, the database 404 contains documents that may be organized into folders 406, 408, 410, and 412. In this example, these documents represent the documents to be published to the system 452 and thus accessed by system 452.

The user-interface module 416 allows the principal of the computer system 402
25 to access and control any of the other modules including without limitation, the memory access module 420, the identify information module 414, and the communication module 422. Although the present invention and in particular, the example shown in Fig. 4 contemplates user interaction on the system 402, such is not necessary to accomplish aspects of this invention.

The memory access module 420 allows the computer system 402 and/or the principal to access data stored on the system, such as data contained in the database 402 and/or data contained in the self-identity information database 424. Alternatively, the computer system 402 may access the memory access module 420 through other
5 modules, such as the identity information document module 414 or the verification and publish module 418. The principal uses the user-interface module 416 to control the memory access module 414.

The identity information document module 414 creates identity information documents, such as the identity information document 300 shown in Fig. 3, by using the
10 memory access module 414 retrieve to pull data from the self-identity information database 424. The identity information document module 414 may create an identity information document upon a command from the principal through the user interface module 416 or may alternatively create identity information documents through a standardized procedure without a direct command from the principal. The identity
15 information document module 414 transfers the identity information document to the communication module 422 for communication to other nodes in the network. In other embodiments, the identity information document is simply stored in memory, and, upon request, the document is transmitted to another system via communication module 422.

The communication module 422 controls communications between the
20 computer system 402 and other nodes of the network, including both sending and receiving information to or from other network nodes, such as computer system 452. In general, the principal controls the communication module 422 through the user-interface module 416 and may through this process command that the identity information document be sent to other computer systems, such as computer system 452.
25 The communication module 422 also allows the principal to send other types of information, such as email, to other nodes of the network.

The verification and publication module 420 receives requests to access and publish documents, such as those in database 404, from other network nodes through the communications module 422. In one embodiment, the verification and publication
30 module 422 performs a gate-keeping function and attempts to verify that the accessing

computer system has permission from the principal to receive the published documents. If permission exists, the verification and publication module 420 uses the memory access module 420 to retrieve the requested documents and publishes the documents by sending them to the communication module 422 for communication to the accessing
5 computer system. If permission does not exist, the verification and publication module 420 denies the request to publish, which is communicated to the accesser through the communication module 422. In an alternative embodiment, the verification and publication module 420 does not act as a gate-keeper, but rather retrieves and publishes all requested documents.

10 Computer system 452 is an example of an accessing node, similar to node 152 of Fig. 1, and contains the following components: 1) a communication module 454; 2) a verification module 460; 3) a storage module 458; 4) a name resolution module 456; and 5) a user-interface module 462.

Similar to the user-interface module 416, the user-interface module 462 allows
15 the principal of the computer system 452 to access and control any of the other modules including without limitation, the communication module 422. So, for example the principal of system 452 might use the user-interface module 416 to direct the communications module 422 to send a request to publish documents to the publishing node 402.

20 Like communication module 422, the communication module 454 controls communications between the computer system 452 and other nodes of the network, including both sending and receiving information from other network nodes, such as computer system 402. The communication module 454 is responsible for forwarding identity information documents that it receives from other network nodes to the
25 verification module 460.

The verification module 460 is responsible for translating identity information documents and determining whether they come from a trusted source, a trusted channel, or can otherwise be authenticated. If the identity information document is accepted, the verification module transfers the identity information document to the storage module
30 458. In essence, the verification module 460 relates to some function of control module

154 in Fig. 1 in that it reads the incoming identity information document 116 and accepts it or rejects it depending upon different variables.

The storage module 458 stores the identity information document for later uses by the computer system 452. As with the verification module 460, the storage module
5 458 also performs some of the functions of the control module 154 shown and described in conjunction with Fig. 1 in that it stores accepted identity information documents.

Finally, the name resolution module 456 is responsible for intercepting a request to access published documents from another network node in the form of a user-friendly handle from the communications module 454. As discussed above, the request most
10 likely comes from the principal via the user-interface module 462. The name resolution module 456 further is responsible for searching the storage module 458 for an identity information document with a principal identifier (such as principal identifier 304 in Fig. 3) that matches the intercepted user-friendly handle, amending the request to substitute or replace the user-friendly handle with the machine location set forth in the identity
15 information document, and sending the request to publish documents in the form of the machine location back to the communication module 454. The name resolution module 456 is similar to resolution module 160 from Fig. 1 in that it that intercepts requests to access documents from an accessing node, translates the user-friendly handle into a machine location for the publishing node, and enters the appropriate information into
20 the web browser of the computer system 152, which, in turn, accesses the requested data on the publishing node.

Fig. 5 shows a method for publishing documents located on a publishing computer system, such as computer system 102 in Fig. 1 or 402 in Fig. 4, to an accessing computer system, such as computer system 152 in Fig. 1 or 452 in Fig. 4.
25 Flow 500 generally relates to the process performed by accessing node.

Flow 500 begins with store operation 502, wherein an identify information document, such as one from a publishing node, is stored in a location on the accessing computer system, such as in a recognized identity database 156 from Fig. 1. Like the identity information document 300 shown in Fig. 3, the identity information document
30 received will contain at a minimum a user-friendly handle identifying the principal of

the publishing system, such as an email address, and a machine location for the principal's computer system, such as an IP address. Sometime thereafter, intercept operation 504 intercepts an initial request to access published documents that are stored on the publishing node. The request may be one initiated by the principal of the
5 accessing node, one initiated by another modules within the accessing computer system, or an automatic request. The request will be in the form of or contain a user-friendly handle identifying the principal of the publishing node or the publishing node itself. For example, the request may be to access documents located at an e-mail address.

Next, search operation 506 searches previously received and stored identity
10 information documents to determine whether a principal identifier matches the user-friendly handle received in the request operation 504. If the system is unable to locate a matching identity information document in step 506, the flow 500 branches NO 507 which notifies the principal or other module that the request has failed. Upon notifying the principal that the request failed, process 500 ends. At this time, the principal may
15 re-enter the request or another, similar request to start the process over.

If instead search operation 506 locates the matching identity information document, i.e., one that has a principal identifier that matches the e-mail address received at step 504, flow 500 branches YES to determination operation 508. Determination operation 508 determines the machine location, which is included in
20 identity information document located in step 506.

Following determination operation 508, resolution operation 510 replaces the user-friendly handle, e.g., the email address with the machine location, e.g., IP address 123.123.023.002. Next, send operation 512 sends the request to publish documents in the form of the machine location, rather than the user-friendly handle, to the publishing
25 node.

Steps 506 and 508-512 happen behind the scenes and are not revealed to the principal. Rather, it appears to the principal that his or her request for access to the published documents is delivered in the form of the user-friendly handle, the email address, and not the meaningless machine location. In this way, a principal may access
30 documents located on a different computer system through knowledge only of a user-

friendly handle identifying the principal of the publishing computer system. The principal of the accessing system does not need to know or remember cumbersome numbers, such as IP addresses, to access these documents.

Fig. 6 likewise shows a method for publishing documents located on a publishing computer system, but shows both the processes performed by the publishing node, such as computer system 102 or 402, and the accessing node, such as computer system 152 or 452.

Flow 600 begins with deliver operation 602, which delivers to another node, such as the accessing node, a path name for a location of one or more documents and a user-friendly handle identifying the principal of the publishing node. For example, send operation might comprise an email message from the principal of the publishing node, whose name is Bob, that says: "check out my photos at bob@xyz.com/photos." Alternatively, the principal may send the path name and user-friendly handle to the accessing node or principal via a telephone call, faxed document, or by some other means. Flow 600 moves to receive operation 604 where the accessing machine receives the path name for documents and user-friendly handle identifying the principal of the publishing machine. Next, send operation 606 sends the accessing node an identity information document for the principal of the publishing machine, in this example, Bob's identity information document. Receive operation 608 receives Bob's identity information document, followed by store operation 609, which stores the identity information document in a recognized identity database, similar to database 156 described and shown in conjunction with Fig. 1.

In an alternative embodiment, verification operation (not shown) occurs after receive operation 608 and before store operation 609. Verification operation attempts to authenticate the principal represented by the identity information document and makes a decision about whether or not to accept and store the identity information document.

Sometime thereafter, intercept operation 610 intercepts an initial request for access to documents. The initial request may originate from a principal or some other source and seeks to gain access to documents at a user-friendly handle/path location,

such as bob@xyz.com/photos. Next, resolution operation 612 resolves the initial request to create an amended request by substituting the user-friendly handle with a machine location using the process described in detail in Fig. 5, namely finding a matching identity information document, determining the machine location, and
5 substituting the machine location for the user-friendly handle.

In an alternative embodiment, parse operation (not shown) separates the user-friendly handle and the path name and searches for a match only with the user-friendly handle portion of the initial request. For example, parse operation would drop “/photos” and search for a principal identifier with Bob’s email at bob@xyz.com.

10 In yet another alternative embodiment, the principal’s identity information document includes more than one machine location and an additional search operation (not shown) search each of the machine locations in the identity information document to determine which machine location contains the path name included in the intercepted request. The resolution operation 612 then substitutes the machine location that
15 corresponds with the path name set forth in the initial request.

After completion of resolution operation 612, send operation 614 sends the amended request for published documents using the machine location instead of the user-friendly handle. If a path name is involved, send operation 614 further includes a request to access documents located at that path.

20 At this point flow 600 shifts back to the publishing machine at receive operation 616, receiving the amended request to access documents at the specified path, in this case, the photos folder, such as folder 118 in Fig. 1. Next, verification operation 618 determines whether the principal of the accessing node is authorized to view the requested documents. If the accessing principal is not authorized, flow 600 branches NO
25 deny operation 620, which denies the request for access to documents and the process ends.

If the accessing principal is authorized, flow 600 branches YES to locate operation 622 which locates the requested documents using the path name included in send operation 614. Finally, publish operation 624 published the requested documents
30 and flow 600 ends.

In one embodiment of the present invention, verification operation 618 checks to determine whether the accessing principal is listed in a recognized identity database, that is, whether the accessing principal has previously sent the publishing node his or her identity information document. If the answer is yes, verification operation 618
5 branches YES to locate operation 622. Thus, in this embodiment a mutual exchange of identity information documents is required for documents to be published by one machine to another. This is the most secure way to publish documents.

In an alternative embodiment of the present invention, verification operation 618 does not require that the accessing principal send its identity information document.
10 Instead, verification operation 618 may consider other variable in determining whether to allow access to the requested documents or verification operation 618 and deny operation 620 may be omitted from flow 600 altogether. Rather, flow 600 would proceed from receive operation 616 directly to location operation 622. In other words, the publishing machine may decide that it will allow anyone with its identity
15 information document to access documents on its system. Although this method is simpler than the last, it is not as secure as requiring a mutual exchange of identity information documents.

In yet another alternative embodiment, notify operation (not shown) notifies the accessing node that the request for access to documents has been denied. Upon
20 notification, another send operation (not shown) may send the publishing node an identity information document representing the accessing node or its principal and start the process over at intercept operation 610. If the publishing node has received the accessing node's identity information document, flow 600 is more likely to branch YES to location operation 622 than to branch NO to deny operation 620.

25 In further alternative embodiment of the present invention, the machine location portion of the identity information document may contain a public key rather than an IP address. As used herein, a "public key" should be construed broadly and be defined as a code unique to a principal consisting of numbers and/or letters with no recognized meaning. The foregoing system and methods would work as described above with one
30 exception, described below.

When a public key is used as a machine location in the identity information document, the public key is used to look up the principal's current machine location, which may be an IP address, using a peer name resolution protocol ("PNRP"). PNRP generally consists of a distributed system of knowledge including hashes of public keys and current machine locations. Thus the public key is used to find a principal's "current" machine location. As used herein, "current machine location" should be construed broadly to mean a machine location at a given point in time. For example, if a computer system uses a dynamic IP address, its current machine location may change from a first point in time to second, later point in time. Use of a public key and PNRP will allow an accessing node to access documents from a publishing node even if the publishing node is using a dynamic IP address.

Alternatively, a current machine location may change if the machine is mobile, such as a lap top computer system or a PDA. PNRP is capable of tracking current machine locations for mobile machines as well. Further, use of a public key and PNRP will allow an accessing node to access documents from a publishing node even if the publishing node is mobile.

If a public key is used as the machine location, the accessing node must know the publishing node's public key in order to gain access to the publishing node's documents. The only way for the accessing node to get the publishing node's public key is for the publishing node to send the accessing node its key. In this way, the publishing node is able to effectively control those who are able to access its resources by controlling to whom it sends its public key.

Fig. 7 shows yet another embodiment of the present invention involving a method for publishing documents located on a publishing computer system, such as computer system 102 or 402, to an accessing computer system, such as computer system 152 or 452.

Flow 700 begins with a register operation registering an encrypted machine name, a host domain, and an associated registered machine location with a conventional DNS server. DNS servers generally store a table with machine names appended to host domains along with a corresponding registered machine location, such as an IP address.

Network users use the DNS server to look up the registered IP addresses for specific machines. As used herein, an "encrypted machine name" shall mean a public key that has been changed into a secret code that people cannot understand or use on normal equipment. One way to create an encrypted machine name is to apply an algorithm to
5 the public key.

Next, deliver operation 706 delivers a path name for a location of one or more documents and a user-friendly handle identifying the principal of the publishing node. Flow 700 moves to receive operation 706 where the accessing machine receives the path name for documents and user-friendly handle identifying the principal of the
10 publishing machine. Sometime thereafter, send operation 708 sends the accessing node an identity information document for the principal of the publishing machine. The identity information document includes at least a principal identifier, a public key, and a domain host for the principal of the publishing node. Receive operation 610 receives the identity information document, followed by store operation 712, which stores the
15 identity information document in a recognized identity database, similar to database 156 described and shown in conjunction with Fig. 1.

Sometime thereafter, intercept operation 712 intercepts an initial request for access to documents. The initial request may originate from a principal or some other source and seeks to gain access to documents at a user-friendly handle/path location.
20 Next, resolution operation 714 resolves the user-friendly handle in the initial request with a public key and domain host set forth in an identity information document with a principal identifier that matches the user-friendly handle using the process described in detail in Fig. 5.

Next, convert operation 718 performs a computation on the public key to
25 convert it to the encrypted machine name and append operation 720 appends the encrypted machine name to the domain host. In one embodiment, the convert operation comprises performing an algorithm on the public key, wherein the algorithm is the same algorithm used by the publishing node in register operation 702. The algorithm may be a standard algorithm used commonly by many network nodes, or may be a specific
30 algorithm that was received by the accessing node in some other way.

Flow 700 then moves to lookup operation 722, which uses the encrypted machine name/domain host combination to look up the registered machine location on a DNS server. After completion of lookup operation 722, amend operation 724 amends the initial request to substitute the user-friendly handle with the registered machine
5 location and sends the amended request for published documents using the registered machine location instead of the user-friendly handle. If a path name is involved, send operation 614 further includes a request to access documents located at that path.

The benefit of using the method illustrated by flow 700 is that it allows a publishing machine to employ additional security measures while utilizing conventional
10 DNS servers. Normally, DNS addresses are publicly available and potential hackers can use the DNS servers to learn a publishing node's machine name and machine location. Hackers may then use this information to gain unauthorized access to a publishing node's resources. Registering an encrypted machine name with the DNS server prevents hackers from learning the publishing node's machine name, but makes
15 it more difficult for authorized user's to gain access to the publishing node's resources. Process 700 allows authorized accessing nodes to access the publishing node's resources with knowledge only of the user-friendly handle. The publishing node does not need to know the unfriendly encrypted machine name or domain host in order to access the publishing node's resources. Security is enhanced because the accessing
20 node must receive the publishing node's identity information document in order to utilize the name resolution method.

Although the invention has been described in language specific to computer structural features, methodological acts and by computer readable media, it is to be understood that the invention defined in the appended claims is not necessarily limited
25 to the specific structures, acts or media described. As an example, different formats other than XML may be used to encode identification information. Therefore, the specific structural features, acts and mediums are disclosed as exemplary embodiments implementing the claimed invention.

Using the above described methods of name resolution, the present invention
30 established that a user-friendly handle identifying the principal of the publishing

machine may be associated with or resolve to a machine location so that the principal of the accessing machine need only know the user-friendly handle in order to access the requested documents on the publishing machine. Further the association process is transparent and occurs without the principal's knowledge or involvement.

- 5 The above specification, examples and data provide a complete description of the manufacture and use of the composition of the invention. Since many embodiments of the invention can be made without departing from the spirit and scope of the invention, the invention resides in the claims hereinafter appended.